

	Type	L #	Hits	Search Text	DBs	Time Stamp
1	BRS	L1	143	replay same load same instruction	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:31
2	BRS	L2	2	1 and invalid near3 flag	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:29
3	BRS	L3	74	1 and invalid	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:31
4	BRS	L4	6	3 and 'executed properly'	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:30
5	BRS	L5	4	4 not 2	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:30
6	BRS	L6	27	replay same load adj instruction same store adj instruction	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:36

7	BRS	L7	9	6 and invalid	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:39
---	-----	----	---	---------------	--	---------------------

	Type	L #	Hits	Search Text	DBs	Time Stamp
8	BRS	L8	5	6 and invalid and checker	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:39
9	BRS	L9	1098	8 nto 7	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:35
10	BRS	L10	0	8 not 7	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:36
11	BRS	L11	140	replay and load adj instruction and store adj instruction	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:39
12	BRS	L12	12	11 and invalid and checker	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:36
13	BRS	L13	12	12 and memory	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:37

14	BRS	L14	7	12 not 8	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:38
----	-----	-----	---	----------	--	---------------------

	Type	L #	Hits	Search Text	DBs	Time Stamp
15	IS&R	L15	2423	((712/244,219,216,225,32) or (711/169)).CCLS.	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:38
16	BRS	L16	68	15 and replay	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:38
17	BRS	L17	37	16 and load adj instruction and store adj instruction	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:39
18	BRS	L18	19	17 and invalid	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:39
19	BRS	L19	7	17 and invalid and checker	US- PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB	2005/05/24 14:39


[Subscribe \(Full Service\)](#) [Register \(Limited Service, Free\)](#) [Login](#)

 Search: ☒ The ACM Digital Library ☐ The Guide

replay and memory and checker and load and store and invalid

SEARCH


[Feedback](#) [Report a problem](#) [Satisfaction survey](#)

Terms used

 replay and memory and checker and load and store and invalid and bus and queue and instruction  
 queue and clear and set and interface and flag

 Found  
 25,468  
 of  
 154,226

 Sort results  
 by

relevance



Save results to a Binder

Try an Advanced Search

 Try this search in [The ACM Guide](#)

 Display  
 results

expanded form



Search Tips

☐ Open results in a new  
 window

Results 1 - 20 of 200

 Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

Best 200 shown

 Relevance scale ☐ ☐ ☐ ☐ ☐

### 1 [The KScalar simulator](#)

J. C. Moure, Dolores I. Rexachs, Emilio Luque

 March 2002 **Journal on Educational Resources in Computing (JERIC)**, Volume 2 Issue 1

 Full text available: [pdf\(493.35 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Modern processors increase their performance with complex microarchitectural mechanisms, which makes them more and more difficult to understand and evaluate. KScalar is a graphical simulation tool that facilitates the study of such processors. It allows students to analyze the performance behavior of a wide range of processor microarchitectures: from a very simple in-order, scalar pipeline, to a detailed out-of-order, superscalar pipeline with non-blocking caches, speculative execution, and comp ...

**Keywords:** Education, pipelined processor simulator

### 2 [Speculative dynamic vectorization](#)

Alex Pajuelo, Antonio González, Mateo Valero

 May 2002 **ACM SIGARCH Computer Architecture News**, Volume 30 Issue 2

 Full text available: [pdf\(1.00 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)  
[Publisher Site](#)

Traditional vector architectures have shown to be very effective for regular codes where the compiler can detect data-level parallelism. However, this SIMD parallelism is also present in irregular or pointer-rich codes, for which the compiler is quite limited to discover it. In this paper we propose a microarchitecture extension in order to exploit SIMD parallelism in a speculative way. The idea is to predict when certain operations are likely to be vectorizable, based on some previous history i ...

**Keywords:** Speculative dynamic vectorization, wide buses, speculative data computation, control independence, vector instructions

### 3 [System support for pervasive applications](#)

Robert Grimm, Janet Davis, Eric Lemar, Adam Macbeth, Steven Swanson, Thomas Anderson, Brian Bershad, Gaetano Borriello, Steven Gribble, David Wetherall

 November 2004 **ACM Transactions on Computer Systems (TOCS)**, Volume 22 Issue 4

 Full text available: [pdf\(1.82 MB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Pervasive computing provides an attractive vision for the future of computing.

Computational power will be available everywhere. Mobile and stationary devices will dynamically connect and coordinate to seamlessly help people in accomplishing their tasks. For this vision to become a reality, developers must build applications that constantly adapt to a highly dynamic computing environment. To make the developers' task feasible, we present a system architecture for pervasive computing, called & ...

**Keywords:** Asynchronous events, checkpointing, discovery, logic/operation pattern, migration, one.world, pervasive computing, structured I/O, tuples, ubiquitous computing

#### 4 Algorithms for scalable synchronization on shared-memory multiprocessors

John M. Mellor-Crummey, Michael L. Scott

February 1991 **ACM Transactions on Computer Systems (TOCS)**, Volume 9 Issue 1

Full text available:  [pdf\(3.07 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Busy-wait techniques are heavily used for mutual exclusion and barrier synchronization in shared-memory parallel programs. Unfortunately, typical implementations of busy-waiting tend to produce large amounts of memory and interconnect contention, introducing performance bottlenecks that become markedly more pronounced as applications scale. We argue that this problem is not fundamental, and that one can in fact construct busy-wait synchronization algorithms that induce no memory or interc ...

#### 5 Tarantula: a vector extension to the alpha architecture

Roger Espasa, Federico Ardanaz, Joel Emer, Stephen Felix, Julio Gago, Roger Gramunt, Isaac Hernandez, Toni Juan, Geoff Lowney, Matthew Mattina, André Seznec

May 2002 **ACM SIGARCH Computer Architecture News**, Volume 30 Issue 2

Full text available:  [pdf\(1.19 MB\)](#)  [Publisher Site](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Tarantula is an aggressive floating point machine targeted at technical, scientific and bioinformatics workloads, originally planned as a follow-on candidate to the EV8 processor [6, 5]. Tarantula adds to the EV8 core a vector unit capable of 32 double-precision flops per cycle. The vector unit fetches data directly from a 16 MByte second level cache with a peak bandwidth of sixty four 64-bit values per cycle. The whole chip is backed by a memory controller capable of delivering over 64 GBytes/s ...

**Keywords:** Vector Processor, Microprocessor, High Performance, Bandwidth, Power, Instruction Set Architecture, Virtual Memory, Cache Coherency

#### 6 The hardware architecture of the CRISP microprocessor

D. R. Ditzel, H. R. McLellan, A. D. Berenbaum

June 1987 **Proceedings of the 14th annual international symposium on Computer architecture**


Full text available:  [pdf\(930.17 KB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

#### 7 Memory Ordering: A Value-Based Approach

Harold W. Cain, Mikko H. Lipasti

March 2004 **ACM SIGARCH Computer Architecture News , Proceedings of the 31st annual international symposium on Computer architecture - Volume 00**, Volume 32 Issue 2

Full text available:  [pdf\(244.36 KB\)](#)

Additional Information: [full citation](#), [abstract](#)


Conventional out-of-order processors employ a multi-ported, fully-associative load queue to guarantee correct memory reference order both within a single thread of execution and across threads in a multiprocessor system. As improvements in process technology and pipelining lead to higher clock frequencies, scaling this complex structure to accommodate a

larger number of in-flight loads becomes difficult if not impossible. Furthermore, each access to this complex structure consumes excessive amounts of e ...

### 8 Half-price architecture

Ilhyun Kim, Mikko H. Lipasti

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available:  pdf(278.61 KB) Additional Information: [full citation](#), [abstract](#), [references](#)

Current-generation microprocessors are designed to process instructions with one and two source operands at equal cost. Handling two source operands requires multiple ports for each instruction in structures--such as the register file and wakeup logic--which are often in the processor's critical timing paths. We argue that these structures are overdesigned since only a small fraction of instructions require two source operands to be processed simultaneously. In this paper, we propose the half-pr ...

### 9 Remote queues: exposing message queues for optimization and atomicity

Eric A. Brewer, Frederic T. Chong, Lok T. Liu, Shamik D. Sharma, John D. Kubiatowicz

July 1995 **Proceedings of the seventh annual ACM symposium on Parallel algorithms and architectures**

Full text available:  pdf(1.78 MB) Additional Information: [full citation](#), [references](#), [citing](#), [index terms](#)

### 10 Runtime Power Monitoring in High-End Processors: Methodology and Empirical Data

Canturk Isci, Margaret Martonosi

December 2003 **Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture**

Full text available:  pdf(921.50 KB) Additional Information: [full citation](#), [abstract](#), [citing](#), [index terms](#)

With power dissipation becoming an increasingly vexing problem across many classes of computer systems, measuring power dissipation of real, running systems has become crucial for hardware and software system research and design. Live power measurements are imperative for studies requiring execution times too long for simulation, such as thermal analysis. Furthermore, as processors become more complex and include a host of aggressive dynamic power management techniques, per-component estimates of powerd ...

### 11 Accelerating shared virtual memory via general-purpose network interface support

Angelos Bilas, Dongming Jiang, Jaswinder Pal Singh

February 2001 **ACM Transactions on Computer Systems (TOCS)**, Volume 19 Issue 1

Full text available:  pdf(178.88 KB) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#), [review](#)

Clusters of symmetric multiprocessors (SMPs) are important platforms for high-performance computing. With the success of hardware cache-coherent distributed shared memory (DSM), a lot of effort has also been made to support the coherent shared-address-space programming model in software on clusters. Much research has been done in fast communication on clusters and in protocols for supporting software shared memory across them. However, the performance of software virtual memory (SVM) is sti ...

**Keywords:** applications, clusters, shared virtual memory, system area networks

### 12 Decoupled hardware support for distributed shared memory

Steven K. Reinhardt, Robert W. Pfile, David A. Wood

May 1996 **ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture**, Volume 24 Issue 2

Full text available:  pdf(1.47 MB) Additional Information: [full citation](#), [abstract](#), [references](#), [citing](#), [index terms](#)

This paper investigates hardware support for fine-grain distributed shared memory (DSM)



in networks of workstations. To reduce design time and implementation cost relative to dedicated DSM systems, we decouple the functional hardware components of DSM support, allowing greater use of off-the-shelf devices. We present two decoupled systems, Typhoon-0 and Typhoon-1. Typhoon-0 uses an off-the-shelf protocol processor and network interface; a custom access control device is the only DSM-specific hard ...

13 The Clipper processor: instruction set architecture and implementation

W. Hollingsworth, H. Sachs, A. J. Smith

February 1989 **Communications of the ACM**, Volume 32 Issue 2

Full text available:  [pdf\(4.67 MB\)](#)


Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#), [review](#)

Intergraph's CLIPPER microprocessor is a high performance, three chip module that implements a new instruction set architecture designed for convenient programmability, broad functionality, and easy future expansion.

14 SPLASH: Stanford parallel applications for shared-memory

Jaswinder Pal Singh, Wolf-Dietrich Weber, Anoop Gupta

March 1992 **ACM SIGARCH Computer Architecture News**, Volume 20 Issue 1

Full text available:  [pdf\(3.04 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [citations](#), [index terms](#)

We present the Stanford Parallel Applications for Shared-Memory (SPLASH), a set of parallel applications for use in the design and evaluation of shared-memory multiprocessing systems. Our goal is to provide a suite of realistic applications that will serve as a well-documented and consistent basis for evaluation studies. We describe the applications currently in the suite in detail, discuss some of their important characteristics, and explore their behavior by running them on a real multiprocess ...

15 Synchronization and communication in the T3E multiprocessor

Steven L. Scott

September 1996 **Proceedings of the seventh international conference on Architectural support for programming languages and operating systems**, Volume 31 , 30 Issue 9 , 5

Full text available:  [pdf\(1.34 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

This paper describes the synchronization and communication primitives of the Cray T3E multiprocessor, a shared memory system scalable to 2048 processors. We discuss what we have learned from the T3D project (the predecessor to the T3E) and the rationale behind changes made for the T3E. We include performance measurements for various aspects of communication and synchronization. The T3E augments the memory interface of the DEC 21164 microprocessor with a large set of explicitly-managed, external r ...

16 Cache Memories

Alan Jay Smith

September 1982 **ACM Computing Surveys (CSUR)**, Volume 14 Issue 3


Full text available:  [pdf\(4.61 MB\)](#)

Additional Information: [full citation](#), [references](#), [citations](#), [index terms](#)

17 A system for authenticated policy-compliant routing

Barath Raghavan, Alex C. Snoeren

August 2004 **ACM SIGCOMM Computer Communication Review , Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications**, Volume 34 Issue 4

Full text available:  [pdf\(219.77 KB\)](#) Additional Information: [full citation](#), [abstract](#), [references](#), [index terms](#)

Internet end users and ISPs alike have little control over how packets are routed outside of their own AS, restricting their ability to achieve levels of performance, reliability, and utility that might otherwise be attained. While researchers have proposed a number of source-

routing techniques to combat this limitation, there has thus far been no way for independent ASes to ensure that such traffic does not circumvent local traffic policies, nor to accurately determine the correct party to char ...

**Keywords:** authentication, capabilities, overlay networks, source routing

18 Analysis of multiprocessor cache organizations with alternative main memory update policies

W. C. Yen, K. S. Fu

May 1981 **Proceedings of the 8th annual symposium on Computer Architecture**

Full text available:  [pdf\(1.04 MB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#), [citations](#), [index terms](#)

Cache memory has played a significant role in the memory hierarchy and has been used extensively in large systems and minisystems. The effectiveness of cache memories with alternative main memory update policies in a multiprocessor system is a major concern in this paper. The performances of write-through with write-allocation or no-write allocation, buffered write-through, flag-swap, and buffered flag-swap policies have been analyzed. Because of the dominating cost of the interface between ...

19 Illustrative risks to the public in the use of computer systems and related technology

Peter G. Neumann

January 1996 **ACM SIGSOFT Software Engineering Notes**, Volume 21 Issue 1

Full text available:  [pdf\(2.54 MB\)](#)

Additional Information: [full citation](#)

20 A "flight data recorder" for enabling full-system multiprocessor deterministic replay

Min Xu, Rastislav Bodik, Mark D. Hill

May 2003 **ACM SIGARCH Computer Architecture News , Proceedings of the 30th annual international symposium on Computer architecture**, Volume 31 Issue 2

Full text available:  [pdf\(311.95 KB\)](#)

Additional Information: [full citation](#), [abstract](#), [references](#)

Debuggers have been proven indispensable in improving software reliability. Unfortunately, on most real-life software, debuggers fail to deliver their most essential feature --- a faithful replay of the execution. The reason is non-determinism caused by multithreading and non-repeatable inputs. A common solution to faithful replay has been to record the non-deterministic execution. Existing recorders, however, either work only for datarace-free programs or have prohibitive overhead. As a step toward ...

Results 1 - 20 of 200

Result page: [1](#) [2](#) [3](#) [4](#) [5](#) [6](#) [7](#) [8](#) [9](#) [10](#) [next](#)

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2005 ACM, Inc.

[Terms of Usage](#) [Privacy Policy](#) [Code of Ethics](#) [Contact Us](#)

Useful downloads:  [Adobe Acrobat](#)  [QuickTime](#)  [Windows Media Player](#)  [Real Player](#)